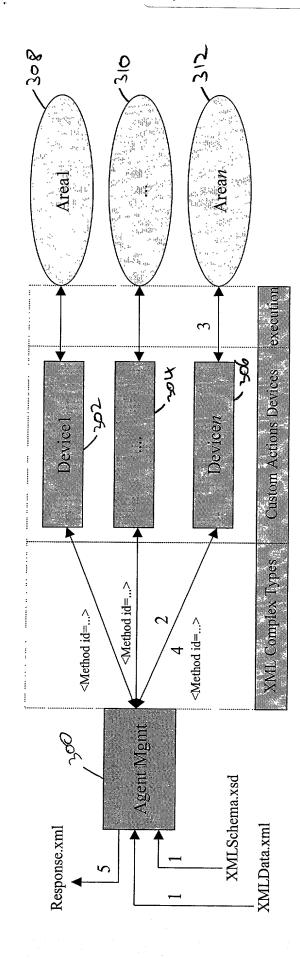
Inventor: NEDBAL, M. SN unknown/Sheet 1 of 27 Atty. Dkt.: 550-320

Architecture overview



Agent Mgmt splits the data into parts and hands the <Method> over to the 1) Request XML file is passed to Agent Mgmt containing complex types corresponding Custom Actions Devices

2) The complex type <Method> describes the method to execute and its parameters

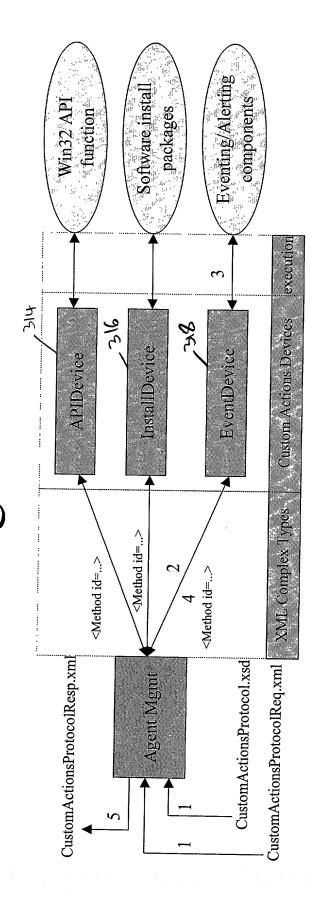
3) Custom Actions Devices execute the functions in a certain area they are responsible for and pack the result into XML complex types

4) Complex types containing result data are returned to Agent Mgmt

5) XML stream has been packed and is returned

Inventor: NEDBAL, M. SN unknown/Sheet 2 of 27 Atty. Dkt.: 550-320

Executing Custom Actions



1) Request XML file is passed to Agent Mgmt containing complex types

2) The complex type <Method> describes the method to execute and its parameters Each <Method> is passed to the corresponding Custom Actions Device

3) Custom Actions Devices execute the functions and pack the result into XML complex types

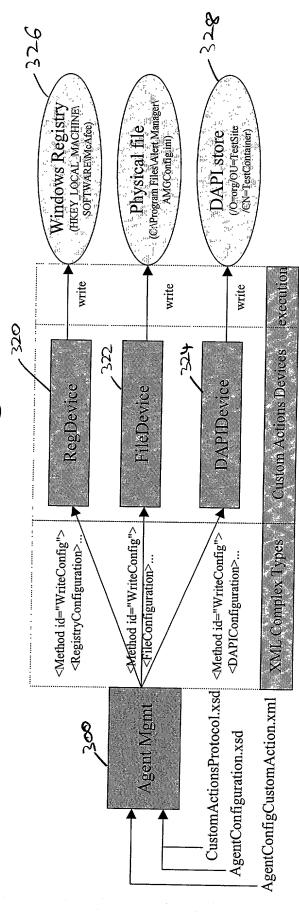
4) Complex types containing result data are returned to Agent Mgmt

5) XML stream has been packed and is returned

1-10-1

Inventor: NEDBAL, M. SN unknown/Sheet 3 of 27 Atty. Dkt.: 550-320

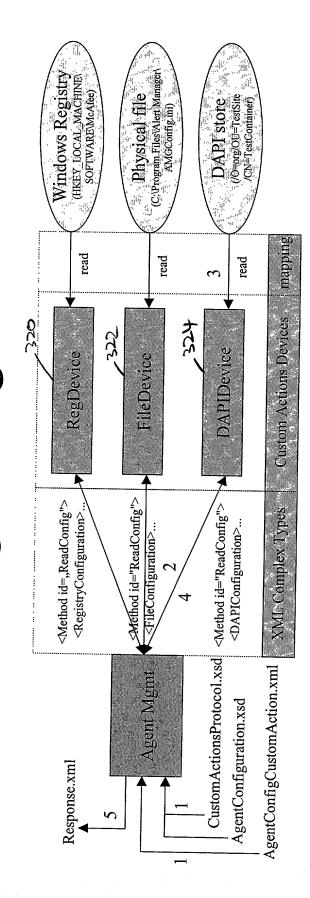
Deploying Configuration Data



- 1) Agent Mgmt receives AgentConfigCustomAction.xml
- 2) .xml file is validated against .xsd file(s) to make sure valid data is written
- 3) XML Complex types are sent to Custom Actions Devices, the parameters containing the configuration data
- 4) Custom Actions Devices update the config store they are responsible for
 - 5) Optionally a return value can be returned as a Response.xml file

17 V. j. Inventor: NEDBAL, M. SN unknown/Sheet 4 of 27 Atty. Dkt.: 550-320

Retrieving Configuration Data



1) Request XML file is passed to Agent Mgmt containing empty requested complex types

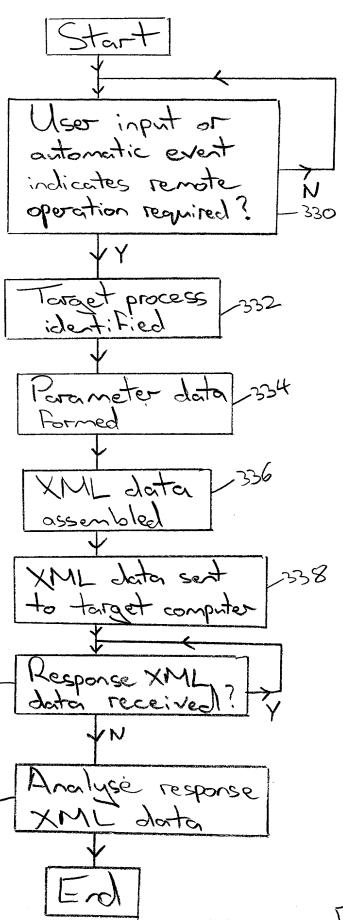
2) The empty complex types are passed to the Custom Actions Device

3) Complex types are filled by mapping required data from store

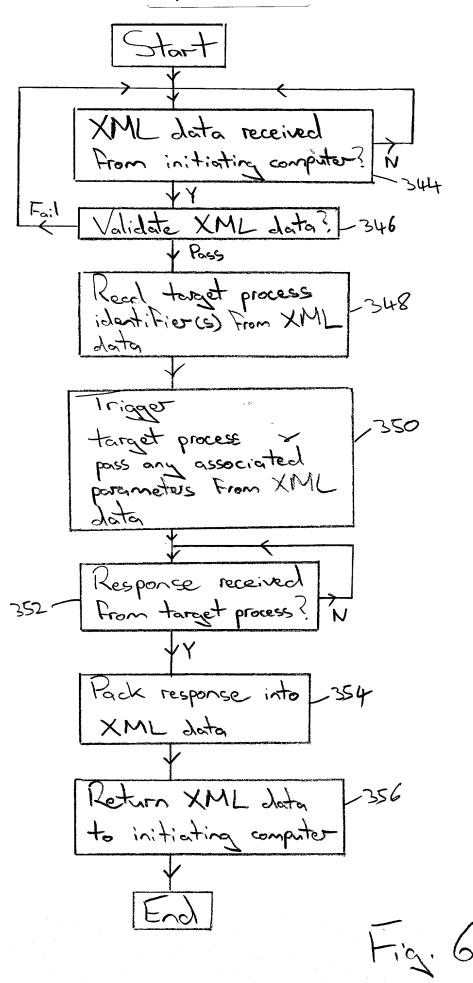
4) Complex types containing data are returned to Agent Mgmt 5) XML stream has been packed and is returned

11

The state of the s



with the set that the set that the set the set that the set that the set the set that the set the set



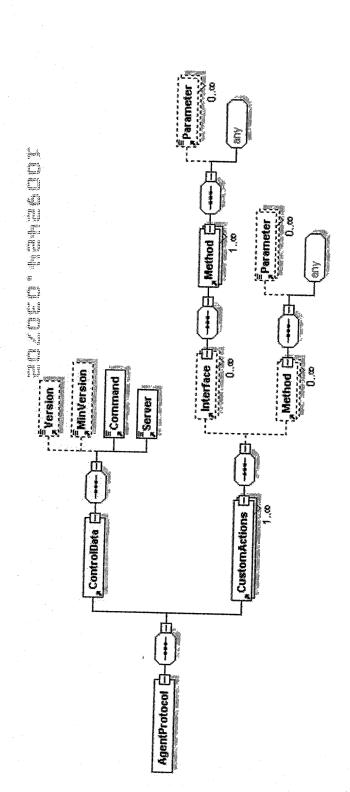
orge process Receive parameter data From agent The state of the s Perform operation موو

Inventor: NEDBAL, M. SN unknown/Sheet 8 of 27 Atty. Dkt.: 550-320

6	V)
1_	5/

	Protocol Specification
<gontroldata></gontroldata>	contains Agent specific control data like version information, the command to execute and computer information sending the custom actions information.
€Command>	the command field could be used to easily determine if data is going to be retrieved (Respond ToCustom Action) - (RequestCustom Action) or returned affer the action has been executed (Respond ToCustom Action) - optional.
<\$er/cep	computer information of the sender. If the Agent is able to process requests in parallel and asynchronosty, this information is useful. The simplest form of this field contains the computer name.
<customactions></customactions>	any number of custom action COM servers or dils or executeables required for the tasks are listed within the CustomActions.
	the id specifies a CLSID if the custom action method is contained in a COM server or the path to a dillexe file if a library or an executeable implements the method to execute. If a CLSID is specified, the following shreface? complex type is required to specify the Interface of the COM server containing the method to execute. Otherwise sinterfaces is not required and any number of sMethod? types follow immediately.
<pre> < interface></pre>	only required if the *Method> is implemented in a COM server.
<method></method>	une interace denumer of the COM server (int); any number of Methods implemented in the custom action device.
[O]	the Method name. In case of a COM server, this is the method name of the COM interface, else this adenotes the name of an executeable.
< Parameter>	any number of parameters required for the method. This includes requested out parameters, which are listed, but don't contain data and inout parameters which have a different value on response than on request.
	the name of the parameter.
lnout	t. Specifies if the parameter is requested, pas
-sainy>	any other non standard XML datatype can follow

Inventor: NEDBAL, M. SN unknown/Sheet 9 of 27 Atty. Dkt.: 550-320



Custom Actions Protocol. xsel

Inventor: NEDBAL, M. SN unknown/Sheet 10 of 27 Atty. Dkt.: 550-320

and the second states of the second s

<?xml version="1.0" ?> - <AgentProtocol xmlns="http://www.nai.com"</pre> xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.nai.com CustomActionsProtocol.xsd"> - <ControlData> <Version>0x01000001</version> <MinVersion>0x01000001</MinVersion> <Command>RequestCustomAction</Command> <Server>ned1wnts2ke</Server> </ControlData> - < Custom Actions id="<AGENT_INSTALLED_DIR>\\CustomActionsLibrary\\CustAct1.dll"> - <Method id="GetRegStringValue"> <Parameter id="Key" type="xs:string"</pre> inout="in"><AGENT_INSTALLED_REGKEY></Parameter> <Parameter id="Valuename" type="xs:string" inout="in">AgentVersion</Parameter> <Parameter id="Result" type="xs:string" inout="out" /> </Method> </CustomActions> - <CustomActions id="{06E0062A-5069-4793-ACED-F80BE1BBC4AF}"> - <Interface id="{C9E1CC03-8007-412A-8F5D-532C57DF4482}"> - <Method id="ExecuteSilentInstallation"> <Parameter id="ProductName" type="xs:string"</pre> inout="in">TestInstallProduct</Parameter> <Parameter id="ProductVersion" type="xs:decimal" inout="in">0x01000001</Parameter> <Parameter id="Location" type="xs:string"</pre> inout="in">c:\InstallImages</Parameter> <Parameter id="Result" type="xs:string" inout="out" /> </Method> </Interface> - <Interface id="{C9E1CC03-8007-412A-8F5D-532C57DF4482}"> - <Method id="GetSystemDirectory"> <Parameter id="Directory" type="xs:string" inout="out" /> <Parameter id="Result" type="xs:decimal" inout="out" /> </Method> </Interface> </CustomActions> - <CustomActions id="{06E0062B-5069-4793-ACED-F80BE1BBC4AF}"> - <Interface id="{A000CC03-8007-412A-8F5D-532C57DF4482}"> - <Method id="TriggerEvent"> <Parameter id="EventID" type="xs:decimal" inout="in">1000</Parameter> <Parameter id="EventDescription" type="xs:decimal"</pre> inout="in">The event %EventID% has been triggered by % USERNAME% on computer %COMPUTERNAME%, The % FILENAME% file is infected with %VIRUSNAME%. This has been detected by engineversion %ENGINEVERSION% datversion %DATVERSION%.</Parameter> <Parameter id="COMPUTERNAME" type="xs:string"</pre> inout="in">sourcecomputer</Parameter> <Parameter id="USERNAME" type="xs:string" inout="in">sourceuser</Parameter> <Parameter id="FILENAME" type="xs:string"</pre> inout="in">kernel32.dll</Parameter> <Parameter id="VIRUSNAME" type="xs:string"</pre>

Fig. 10A

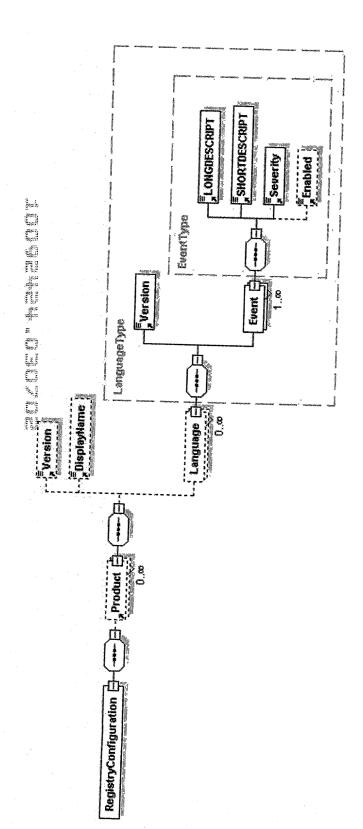
Inventor: NEDBAL, M. SN unknown/Sheet 11 of 27 Atty. Dkt.: 550-320

The limit and had been some the from the first the first

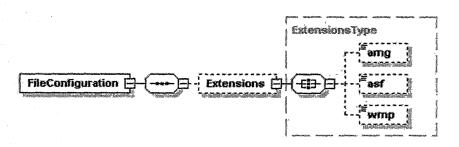
Inventor: NEDBAL, M. SN unknown/Sheet 12 of 27 Atty. Dkt.: 550-320

```
<?xml version="1.0" ?>
  - <AgentProtocol xmlns="http://www.nai.com"</p>
     xmins:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:schemaLocation="http://www.nai.com CustomActionsProtocol.xsd">
   - <ControlData>
       <Version>0x01000001</Version>
       <MinVersion>0x01000001</MinVersion>
       <Command>RspondToCustomAction</Command>
       <Server>ned1wnts2ke</Server>
     </ControlData>
   - < Custom Actions
       id="<AGENT_INSTALLED_DIR>\\CustomActionsLibrary\\CustAct1.dll">
     - <Method id="GetRegStringValue">
        <Parameter id="Result" type="xs:string"
          inout="out">5.0.1.10</Parameter>
       </Method>
     </CustomActions>
   - <CustomActions id="{06E0062A-5069-4793-ACED-F80BE1BBC4AF}">
     - <Interface id="{C9E1CC03-8007-412A-8F5D-532C57DF4482}">
       - <Method id="ExecuteSilentInstallation">
          <Parameter id="Result" type="xs:string" inout="out">Error: Invalid
            Image path specified.</Parameter>
        </Method>
- 12 m
       </Interface>
ħJ
     - <Interface id="{C9E1CC03-8007-412A-8F5D-532C57DF4482}">
      - <Method id="GetSystemDirectory">
          <Parameter id="Directory" type="xs:string"</pre>
            inout="out">C:\Winnt\System32</Parameter>
          <Parameter id="Result" type="xs:decimal"
            inout="out">0</Parameter>
        </Method>
       </Interface>
     </CustomActions>
   - <CustomActions id="{06E0062B-5069-4793-ACED-F80BE1BBC4AF}">
     - <Interface id="{A000CC03-8007-412A-8F5D-532C57DF4482}">
      - <Method id="TriggerEvent">
          <Parameter id="Result" type="xs:string" inout="out">Event sent to
           testcomputer2</Parameter>
        </Method>
      </Interface>
     </CustomActions>
   </AgentProtocol>
```

7



RegistryCoutiporation i Agent Coutigonation.xsol



Agent Configuration.xsd - File Configuration

Agent Configuration. xsol - DAPI Configuration

- <Extensions>

Agent Config Custom Action. xul

Inventor: NEDBAL, M. SN unknown/Sheet 16 of 27 Atty. Dkt.: 550-320

```
<?xml version="1.0" ?>
- <AgentProtocol xmlns="http://www.nai.com"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.nai.com CustomActionsProtocol.xsd
   http://www.nai.com AgentConfiguration.xsd">
 - <ControlData>
     <Version>0x01000001</Version>
     <MinVersion>0x01000001</MinVersion>
     <Command>RequestCustomAction</Command>
     <Server>ned1wnts2ke</Server>
   </ControlData>
 - <CustomActions id="RegistryMapping.dll">
   - <Method id="WriteConfig">

    - <RegistryConfiguration</li>

        id="HKEY LOCAL MACHINE\SOFTWARE\McAfee">
      - < Product id = "Alert Manager">
          <Version>0x04070000</Version>
          <DisplayName>Alert Manager 4.7/DisplayName>
        - <Language id="0407">
           <Version>0x01000002</Version>
          - <Event id="1">
             <LONGDESCRIPT>Das ist eine Test-Nachricht von Alert
               Manager.</LONGDESCRIPT>
             <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
             <Severity>5</Severity>
             <Enabled>1</Enabled>
           </Event>
          </Language>
        - <Language id="0409">
           <Version>0x01000002</Version>
          - <Event id="1">
             <LONGDESCRIPT>This is an alert manager test
               messae. </LONGDESCRIPT>
             <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
             <Severity>0</Severity>
             <Enabled>1</Enabled>
           </Event>
          - <Event id="2">
             <LONGDESCRIPT>Text of event 2.</LONGDESCRIPT>
             <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
             <Severity>1</Severity>
           </Event>
         </Language>
        </Product>
      </RegistryConfiguration>
    </Method>
   - < Method id="ReadConfig">
      < Registry Configuration
        id="HKEY_LOCAL_MACHINE\SOFTWARE\McAfee\*"/>
    </Method>
   </CustomActions>
 - <CustomActions id="INIFileMapping.dll">
   - <Method id="WriteConfig">
    - <FileConfiguration id="C:\Program Files\Alert
        Manager\AMGConfig.ini">
```

Fig. 15A

Agent Config Custom Action.xul

Inventor: NEDBAL, M. SN unknown/Sheet 17 of 27 Atty. Dkt.: 550-320

```
<amg>AMGConfig</amg>
        <asf>MPEGVideo</asf>
        <wmp>MPEGVideo2</wmp>
       </Extensions>
     </FileConfiguration>
   </Method>
 - <Method id="ReadConfig">
     <FileConfiguration id="C:\Program Files\Alert</pre>
       Manager\AMGConfig.ini" />
   </Method>
 </CustomActions>
- < CustomActions id="MAPIMapping.dll">
 - <Method id="WriteConfig">
   - <DAPIConfiguration id="/O=org/OU=TestSite/CN=TestContainer">
      <BinaryProperty>0123456789ABCDEF00000</BinaryProperty>
     </DAPIConfiguration>
   </Method>
 - <Method id="ReadConfig">
     <DAPIConfiguration id="/O=org/OU=TestSite/CN=TestContainer" />
   </Method>
 </CustomActions>
</AgentProtocol>
```



User input or required? a complex data type execution process for operation parameter data XML data

with the first of the first fam the first of the first of

star Receive XML data from source computer 374 Parse XML data to identify complex type(s) Match identified complex type(s) with execution process (es) I rigger execution 380 processies) including passing parameter data

Inventor: NEDBAL, M. SN unknown/Sheet 20 of 27 Atty. Dkt.: 550-320

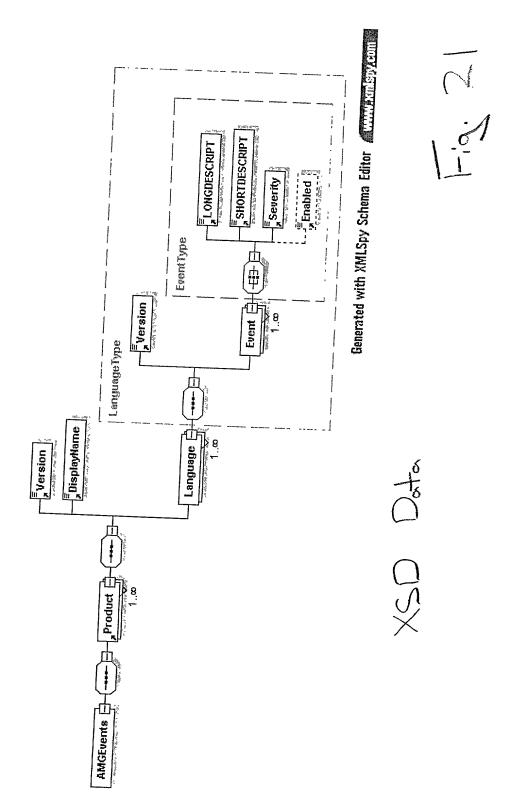
X D			or or or or		v 2) (•	₩ w						
		Data	(value not set)	·	This is an alert manager test messge.	0	Testing			である。 マー・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	d="0409"(Event.id="1"	1.8.10 00	,
Marie de Constantante de Constanta Constanta de Constanta		Type	REG_5Z	REG_SZ	REG_SZ	REG_52	REG_5Z				eelAlert.Manager/Events\Product.id="Alert Manager"(Language id="0409"(Event.id="1"	ata	
		Name	(Default)	Enabled	(ab)LONGDESCRIPT	ab) Severity	(1) SHORTDESCRIPT		2000		anager/Events/Product.id=	Registry Data	
💇 Registry Editor	Registry, Edit. View Favorites Help.	Current/Version	B- Devices	Events	Product Id="Riest Manager"		CYCIT IN THE TABLE	Eventide 1	Event id="3"	Event id="4"	I/I/K.Computer/HKEY/LOGAL/MACHINE\SOFTWARE\McARe\AlertM		

al Messages Editor - Alext Manages File Property File 국 조조 출 발기 약 😿	☐ △ Mert Manager ☐ ◆ German (Germany) ☐ ← A 1: Testing ☐ ← A 1: Testing ← A 2: Testing ← A 3: Testing ← A 4: Testing ← A 4: Testing	For Help, press F1
Messages configuration	Message: Property Name Langle Long description Langle Severity Langle Enabled	
	Message Property Value	
		**

Dom Data View

```
<?xml version="1.0" ?>
- <AMGEvents xmlns="http://www.nai.com"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xsi:schemaLocation="http://www.nai.com AMGEvents.xsd">
 - <Product id="Alert Manager">
    <Version>0x04070000</Version>
    <DisplayName>Alert Manager 4.7/DisplayName>
   - <Language id="0407">
      <Version>0x01000002</Version>
    - <Event id="1">
        <LONGDESCRIPT>Das ist eine Test-Nachricht von Alert
         Manager.</LONGDESCRIPT>
        <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
        <Severity>5</Severity>
        <Enabled>1</Enabled>
      </Event>
    </Language>
   - <Language id="0409">
      <Version>0x01000002</Version>
    - <Event id="1">
        <LONGDESCRIPT>This is an alert manager test
         messge.</LONGDESCRIPT>
        <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
       <Severity>0</Severity>
       <Enabled>1</Enabled>
      </Event>
    - <Event id="2">
       <LONGDESCRIPT>Text of event 2.</LONGDESCRIPT>
       <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
       <Severity>1</Severity>
      </Event>
    - <Event id="3">
       <LONGDESCRIPT>Text of event 3.</LONGDESCRIPT>
       <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
       <Severity>1</Severity>
      </Event>
    - <Event id="4">
       <LONGDESCRIPT>Text of event 4.</LONGDESCRIPT>
       <SHORTDESCRIPT>Testing</SHORTDESCRIPT>
       <Severity>1</Severity>
      </Event>
    </Language>
  </Product>
 </AMGEvents>
```

XML Data



```
<?xml version="1.0" encoding="UTF-8" ?>
 <!-- edited with XML Spy v4.0.1 U (http://www.xmlspy.com) by Napalm
 (Napalm)
- <xs:schema targetNamespace="http://www.nai.com"</p>
   xmlns="http://www.nai.com"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   elementFormDefault="qualified">
   <xs:element name="DisplayName" type="xs:string" />
   <xs:element name="Enabled" type="xs:boolean" />
 - <xs:complexType name="EventType">
   - <xs:all>
      <xs:element ref="LONGDESCRIPT" />
      <xs:element ref="SHORTDESCRIPT" />
      <xs:element ref="Severity" />
      <xs:element ref="Enabled" minOccurs="0" />
    </xs:all>
    <xs:attribute name="id" type="xs:string" use="required" />
   </xs:complexType>
 - <xs:complexType name="LanguageType">
  - <xs:sequence>
      <xs:element ref="Version" />
      <xs:element name="Event" type="EventType"</pre>
        maxOccurs="unbounded" />
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required" />
  </xs:complexType>
 - <xs:element name="Product">
  - <xs:complexType>
    - <xs:sequence>
        <xs:element ref="Version" />
        <xs:element ref="DisplayName" />
        <xs:element name="Language" type="LanguageType"</pre>
         maxOccurs="unbounded" />
      </xs:sequence>
      <xs:attribute name="id" type="xs:string" use="required" />
    </xs:complexType>
  </xs:element>
 - <xs:element name="AMGEvents">
  - <xs:complexType>
    - <xs:sequence>
        <xs:element ref="Product" maxOccurs="unbounded" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="LONGDESCRIPT" type="xs:string" />
  <xs:element name="SHORTDESCRIPT" type="xs:string" />
  <xs:element name="Severity" type="xs:string" />
  <xs:element name="Version" type="xs:string" />
</xs:schema>
```

XSD Data

Inventor: NEDBAL, M. SN unknown/Sheet 24 of 27 Atty. Dkt.: 550-320

1-19. 24

